UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/091,237	03/04/2002	Hong Su	10013661-1	7188

7590 11/28/2005

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, CO 80527-2400

EXAMINER	
STEVENS, ROBERT	
ART UNIT	**PAPER NUMBER**
2176	

DATE MAILED: 11/28/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	10/091,237	SU ET AL.
	Examiner	Art Unit
	Robert M. Stevens	2176

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *17 October 2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-26* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-26* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This action is responsive to communications: after final amendment filed 10/17/2005 by Su et al. (Application entitled "Transform between source and target xml schemas").

2. The Office withdraws all previous objections / rejections raised in the prior action (mailed 10/17/2005) without further prejudice as to the applicability of the previously presented prior art.

3. Applicant's declarations under 37 CFR 1.131 have been considered, but are deemed to have been untimely presented and therefore ineffective to overcome the Jeong reference at this time. See discussion below.

4. The Office has re-opened prosecution, however, as a result of becoming aware of additional, relevant prior art.

5. Claims 1-26 are pending. Claims 1, 10 and 18 are independent.

Specification

6. The specification is objected to because a Federally Sponsored Research and Development statement, as set forth in MPEP §310, appears to be necessary. On page 29 of the power point briefing [Su, Hong, et al., "Automating Transformation of XML Documents", WIDM 2001 Powerpoint Presentation, Atlanta, Ga, Nov. 2001, pp. 1-29] presented at the WIDM 2001 Conference by the inventors, mention is given to the involvement of the NSF (i.e., a US Government agency that provides R&D funding to researchers). Note that this briefing discloses the subject matter of the WIDM 2001 paper that is the subject of the inventors' Rule 131 affidavits concerning reduction to practice.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8. **Claims 1, 3-4 and 10-16 are rejected under 35 U.S.C. 103(a)** as being unpatentable over Hong Su et al. ("Identification of Syntactically Similar DTD Elements for Schema Matching", The Second International Conference on Web-Age Information Management (Waim 2001), Xi'an, China, July 2001, pp. 1-13, hereafter referred to as

"SchemaMatching") in view of Hong Su et al. ("XEM: Managing the Evolution of XML Documents", Eleventh International Workshop on Research Issues in Data Engineering (RIDE 2001), Heidelberg, Germany, April 1-2, 2001, pp. 1-8, hereafter referred to as "XEM").

Regarding independent claim 1:

SchemaMatching discloses:

> *A method of document transformation comprising:*
> *a) modeling a source XML document corresponding source schema source tree having a plurality of source nodes;* (Abstract, noting discussion of DTD graphs, and page 2, especially customer DTD1)
> *b) modeling a target XML document corresponding target schema nodes;* (Abstract, noting discussion of DTD graphs, and page 2, especially client DTD2) *and*
> *c) ... .*

However, SchemaMatching does not explicitly disclose:

> *... :*
> *a) ... ;*
> *b) ... ; and*
> *c) generating a sequence of transformation operations that transforms said source tree to said target tree.*

XEM, though, discloses:

> *... :*
> *a) ... ;*
> *b) ... ; and*
> *c) generating a sequence of transformation operations that transforms said source tree to said target tree.* (pp. 6-7 section entitled "4. Completeness of DTD Change Operations")

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of XEM for the benefit of SchemaMatching, because to do so would have allowed a designer to change a DTD without requiring change of underlying XML data, as taught by XEM in top left paragraph of page 2. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 3, SchemaMatching further discloses:

> *wherein c) comprises: matching said plurality of source nodes to said plurality of target nodes.* (Abstract and pp. 5-6 section entitled "3.1 Initial Leaf Vertices Matching")

Regarding claim 4, which is dependent upon claim 1, the limitations of claim 1 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *wherein c) comprises: automatically generating said sequence of transformation operations.*

XEM, though, discloses:

> *wherein c) comprises: automatically generating said sequence of transformation operations.* (p. 4 section "3.2 DTD Change Primitives", it being noted that automation is inherent/implicit in any computer-based process)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of XEM for the benefit of SchemaMatching, because to do so would have allowed a designer to change a DTD without requiring change of

underlying XML data, as taught by XEM in top left paragraph of page 2. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding independent claim 10:

SchemaMatching discloses:

> *A method of document transformation comprising:*
> *a) modeling a source schema of XML and a target schema of XML as a tree structure creating a source tree and a target tree, source tree having a plurality of source nodes, said target tree having a plurality of target nodes;* (Abstract, noting discussion of DTD graphs, and page 2, especially customer DTD1 and client DTD2) *and*
> *b) ... , wherein said plurality of source nodes of said source schema are matched and transformed to said plurality of target nodes in said target schema.* (Abstract and pp. 5-6 section entitled "3.1 Initial Leaf Vertices Matching")

However, SchemaMatching does not explicitly disclose:

> *... :*
> *a) ... ; and*
> *b) generating a sequence of transformation operations that transforms said source XML document to said target XML document, ...*

XEM, though, discloses:

> *... :*
> *a) ... ; and*
> b) generating a sequence of transformation operations that transforms said source XML document to said target XML document, ... (pp. 6-7 section entitled "4. Completeness of DTD Change Operations")

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of XEM for the benefit of SchemaMatching, because to do so would have allowed a designer to change a DTD without requiring change of

underlying XML data, as taught by XEM in top left paragraph of page 2. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 11, SchemaMatching further discloses:

> *wherein b) comprises:*
> *b1) each source node in said source tree, selecting a plurality of candidate nodes in said target tree that are possible matches;* (page 2, especially #3 discussing matching)
> *b2) for each source node in said source tree, generating a plurality of node transformation operations transforming to each of said plurality of candidate nodes;* (page 2 #4 re: "best matching plan") *and*
> *b3) for each source node in said source tree, selecting one of said plurality of node transformation operations forming a selected node transformation operation having the least cost of data loss.* (pp. 6-7 section entitled "3.2 Element Graph's Distance", especially noting "Example 4" and the paragraph above "Example 4" which discuss, inter alia, which discuss distance [i.e., cost] and element overlap)

Regarding claim 12, which is dependent upon claim 11, the limitations of claim 11 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *combining said selected node transformation operation for each of said source nodes matched to a target node a sequence of transformation operations that transforms said source schema to said target schema.*

XEM, though, discloses:

> *combining said selected node transformation operation for each of said source nodes matched to a target node a sequence of transformation operations that transforms said source schema to said target schema.* (pp. 6-7 section entitled "4. Completeness of DTD Change Operations")

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of XEM for the benefit of SchemaMatching, because to do so would have allowed a designer to change a DTD without requiring change of underlying XML data, as taught by XEM in top left paragraph of page 2. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 13, which is dependent upon claim 10, the limitations of claim 10 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *wherein said source schema is a source document type definition (DTD) and said target schema a target DTD.*

XEM, though, discloses:

> *wherein said source schema is a source document type definition (DTD) and said target schema a target DTD.* (Abstract)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of XEM for the benefit of SchemaMatching, because to do so would have allowed a designer to change a DTD without requiring change of underlying XML data, as taught by XEM in top left paragraph of page 2. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 14, SchemaMatching further discloses:

> *folding nodes in said source and target trees preprocessing phase to find one-to-one node matching.* (pp. 2-3 section entitled "2.1 Simplification Transformation on DTD")

Regarding claim 15, SchemaMatching further discloses:

> *merging nodes in said source and target trees preprocessing phase to find one-to-one node matching.* (pp. 2-3 section entitled "2.1 Simplification Transformation on DTD")

Regarding claim 16, which is dependent upon claim 10, the limitations of claim 10 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *performing transformation operations only once at a tree and said target tree with the following exceptions:*
>
> > *a relabel operation following an unfold operation;*
> >
> > *said unfold operation following said relabel operation;*
> >
> > *said relabel operation performed between an attribute and an element following or followed by a deletion or an addition of a qmark quantifier node.*

XEM, though, discloses:

> *performing transformation operations only once at a tree and said target tree with the following exceptions:*
>
> > *a relabel operation following an unfold operation;* (pp. 4-6 section entitled "3.2 DTD Change Primitives" in context of p. 3 section entitled "2. Constraint Node")
> >
> > *said unfold operation following said relabel operation;* (pp. 4-6 section entitled "3.2 DTD Change Primitives" in context of p. 3 section entitled "2. Constraint Node")
> >
> > *said relabel operation performed between an attribute and an element following or followed by a deletion or an addition of a*

> *qmark quantifier node.* (pp. 4-6 section entitled "3.2 DTD Change Primitives" in context of p. 3 section entitled "2. Constraint Node")

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of XEM for the benefit of SchemaMatching, because to do so would have allowed a designer to change a DTD without requiring change of underlying XML data, as taught by XEM in top left paragraph of page 2. These references were all applicable to the same field of endeavor, i.e., XML programming.

9. **Claims 2 and 17-21 are rejected under 35 U.S.C. 103(a)** as being unpatentable over Hong Su et al. ("Identification of Syntactically Similar DTD Elements for Schema Matching", The Second International Conference on Web-Age Information Management (Waim 2001), Xi'an, China, July 2001, pp. 1-13, hereafter referred to as "SchemaMatching") in view of Hong Su et al. ("XEM: Managing the Evolution of XML Documents", Eleventh International Workshop on Research Issues in Data Engineering (RIDE 2001), Heidelberg, Germany, April 1-2, 2001, pp. 1-8, hereafter referred to as "XEM") and further in view of Swamy et al. (US Patent No. 6,874,141, filed Jun. 29, 2000 and issued Mar. 29, 2005, hereafter referred to as "Swamy").

Regarding claim 2, which is dependent upon claim 1, the limitations of claim 1 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *d) converting said sequence of transformation operations into an Extensible Stylesheet Language for Transformations (XSLT) script.*

Swamy, though, discloses:

> *d) converting said sequence of transformation operations into an Extensible Stylesheet Language for Transformations (XSLT) script.* (Abstract)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Swamy for the benefit of SchemaMatching in view of XEM, because to do so would have allowed one to compile a graphical representation of data transformations into an XSL stylesheet representation of the mapping, as taught by Swamy in col. 3 lines 19-25. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 17, this claim is substantially similar to claim 2, and therefore likewise rejected.

Regarding independent claim 18:

SchemaMatching discloses:

> *A computer system comprising:*
> *... ; and*
> *... :*
> *modeling a source document corresponding to a source schema as a source tree having a plurality of source nodes;* (Abstract, noting discussion of DTD graphs, and page 2, especially customer DTD1)

> *modeling a target document corresponding to a target schema as a target tree having a plurality of target nodes;* (Abstract, noting discussion of DTD graphs, and page 2, especially client DTD2) *and*
>
> *... .*

However, SchemaMatching does not explicitly disclose:

> *... :*
>
> *... ; and*
>
> *... :*
>
> *... ;*
>
> *... ; and*
>
> *generating a sequence of transformation operations that transforms said source tree to said target tree.*

XEM, though, discloses:

> *... :*
>
> *... ; and*
>
> *... :*
>
> *... ;*
>
> *... ; and*
>
> *generating a sequence of transformation operations that transforms said source tree to said target tree.* (pp. 6-7 section entitled "4. Completeness of DTD Change Operations")

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of XEM for the benefit of SchemaMatching, because to do so would have allowed a designer to change a DTD without requiring change of underlying XML data, as taught by XEM in top left paragraph of page 2. These references were all applicable to the same field of endeavor, i.e., XML programming.

Furthermore, SchemaMatching does not explicitly disclose:

... :

a processor; and

a computer readable memory coupled to said processor and containing program instructions that, when executed, implement a method of document transformation comprising:

... ;

... ; *and*

... .

Swamy, though, discloses:

... :

a processor; (Fig. 15 #521) *and*

a computer readable memory coupled to said processor and containing program instructions that, when executed, implement a method of document transformation (Fig. 15 #527, #529, #531) *comprising:*

... ;

... ; *and*

... .

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Swamy for the benefit of SchemaMatching in view of XEM, because to do so would have allowed one to compile a graphical representation of data transformations into an XSL stylesheet representation of the mapping, as taught by Swamy in col. 3 lines 19-25. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 19, this claim is substantially similar to claim 2, and therefore likewise rejected.

Regarding claim 20, SchemaMatching further discloses:

> *wherein c) in said method comprises: matching said plurality of source nodes to said plurality of target nodes.* (Abstract and pp. 5-6 section entitled "3.1 Initial Leaf Vertices Matching")

Regarding claim 21, which is dependent upon claim 18, the limitations of claim 18 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *wherein c) in said method comprises: automatically generating said sequence of transformation operations.*

Swamy, though, discloses:

> *wherein c) in said method comprises: automatically generating said sequence of transformation operations.* (p. 4 section "3.2 DTD Change Primitives", it being noted that automation is inherent/implicit in any computer-based process)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Swamy for the benefit of SchemaMatching in view of XEM, because to do so would have allowed one to compile a graphical representation of data transformations into an XSL stylesheet representation of the mapping, as taught by Swamy in col. 3 lines 19-25. These references were all applicable to the same field of endeavor, i.e., XML programming.

10. **Claims 5-9 are rejected under 35 U.S.C. 103(a)** as being unpatentable over Hong Su et al. ("Identification of Syntactically Similar DTD Elements for Schema Matching", The Second International Conference on Web-Age Information Management (Waim 2001), Xi'an, China, July 2001, pp. 1-13, hereafter referred to as "SchemaMatching") in view of Hong Su et al. ("XEM: Managing the Evolution of XML Documents", Eleventh International Workshop on Research Issues in Data Engineering (RIDE 2001), Heidelberg, Germany, April 1-2, 2001, pp. 1-8, hereafter referred to as "XEM") and further in view of Peter Buneman et al ("UnQL: A Query Language and Algebra for SemiStructured Data Based on Structural Recursion", The VLDB Journal, Issue No. 9, Springer-Verlag, © 2000, pp. 76-110, hereafter referred to as "Buneman").

Regarding claim 5, which is dependent upon claim 1, the limitations of claim 1 have been previously addressed.

SchemaMatching further discloses:

> *d) for each source node in said source schema, selecting a plurality of candidate nodes in said target schema that are possible matches; each source node in said source schema;* (page 2 #3 re: "matching likelihood of component pairs)
> *e) generating a transforming to each of said plurality of candidate nodes;* (page 2 #4 teaches a "best matching plan") *and*
> *f) ... .*

However, SchemaMatching does not explicitly disclose:

> *d) ... ;*
> *e) ... ; and*
> *f) for each source node in said source schema, selecting one of said plurality of node transformation sequences, a selected node*

transformation sequence, for said sequence of transformation operations that is associated with a least cost of data loss.

Buneman, though, discloses:

> *d) ... ;*
> *e) ... ; and*
> *f) for each source node in said source schema, selecting one of said plurality of node transformation sequences, a selected node transformation sequence, for said sequence of transformation operations that is associated with a least cost of data loss.* (page 88 section entitled "4.1 Value Equivalence")

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Buneman for the benefit of SchemaMatching in view of XEM, because to do so would have allowed one to implement a value-based semistructured data model, as taught by Buneman in last paragraph on page 77. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 6, which is dependent upon claim 5, the limitations of claim 5 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *a match between a source node and a target node, selecting said selected node transformation sequence to achieve a match, where a first cost of data loss for said match is less than a second cost of data when deleting information contained in said source node, in a first iteration of matching.*

Buneman, though, discloses:

> *a match between a source node and a target node, selecting said selected node transformation sequence to achieve a match, where a first*

> *cost of data loss for said match is less than a second cost of data when deleting information contained in said source node, in a first iteration of matching.* (page 88 section entitled "4.1 Value Equivalence")

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Buneman for the benefit of SchemaMatching in view of XEM, because to do so would have allowed one to implement a value-based semistructured data model, as taught by Buneman in last paragraph on page 77. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 7, which is dependent upon claim 6, the limitations of claim 6 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *matching said source node to said target node having a synonymous label to achieve said match.*

XEM, though, discloses:

> *matching said source node to said target node having a synonymous label to achieve said match.* (page 3 section entitled "2.2 The DTD Data Model" discusses node labelling)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of XEM for the benefit of SchemaMatching in view of Buneman, because to do so would have allowed a designer to change a DTD without requiring change of underlying XML data, as taught by XEM in top left paragraph of

page 2. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 8, which is dependent upon claim 5, the limitations of claim 5 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *wherein f) further comprises: a match between a selecting said selected node transformation sequence when an associated cost of data loss is less than a second data loss when deleting source information contained in said source node and adding target information in said target node, second iteration of matching.*

Buneman, though, discloses:

> *wherein f) further comprises: a match between a selecting said selected node transformation sequence when an associated cost of data loss is less than a second data loss when deleting source information contained in said source node and adding target information in said target node, second iteration of matching.* (page 88 section entitled "4.1 Value Equivalence")

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Buneman for the benefit of SchemaMatching in view of XEM, because to do so would have allowed one to implement a value-based semistructured data model, as taught by Buneman in last paragraph on page 77. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 9, which is dependent upon claim 5, the limitations of claim 5 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *wherein f) further comprises: selecting said selected node transformation sequence of data loss.*

Buneman, though, discloses:

> *wherein f) further comprises: selecting said selected node transformation sequence of data loss.* (page 88 section entitled "4.1 Value Equivalence")

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Buneman for the benefit of SchemaMatching in view of XEM, because to do so would have allowed one to implement a value-based semistructured data model, as taught by Buneman in last paragraph on page 77. These references were all applicable to the same field of endeavor, i.e., XML programming.

11. **Claims 22-26 are rejected under 35 U.S.C. 103(a)** as being unpatentable over Hong Su et al. ("Identification of Syntactically Similar DTD Elements for Schema Matching", The Second International Conference on Web-Age Information Management (Waim 2001), Xi'an, China, July 2001, pp. 1-13, hereafter referred to as "SchemaMatching") in view of Hong Su et al. ("XEM: Managing the Evolution of XML Documents", Eleventh International Workshop on Research Issues in Data Engineering (RIDE 2001), Heidelberg, Germany, April 1-2, 2001, pp. 1-8, hereafter referred to as

"XEM") and further in view of Swamy et al. (US Patent No. 6,874,141, filed Jun. 29, 2000 and issued Mar. 29, 2005, hereafter referred to as "Swamy") and Peter Buneman et al ("UnQL: A Query Language and Algebra for SemiStructured Data Based on Structural Recursion", The VLDB Journal, Issue No. 9, Springer-Verlag, © 2000, pp. 76-110, hereafter referred to as "Buneman").

Regarding claim 22, which is dependent upon claim 18, the limitations of claim 18 have been previously addressed.

SchemaMatching further discloses:

> *d) for each source node said source schema, selecting a plurality of schema that are possible matches;* (page 2 #3 re: "matching likelihood of component pairs)
> *e) for each source node said source schema, transforming to each of said plurality of candidate nodes;* (page 2 #4 teaches a "best matching plan") *and*
> *f) ... .*

However, SchemaMatching does not explicitly disclose:

> *d) ... ;*
> *e) ... ; and*
> *f) for each source node in said source schema, selecting one of said plurality of node transformation sequences, a selected node transformation sequence, said associated with a least cost based on an information capacity cost criteria.*

Buneman, though, discloses:

> *d) ... ;*
> *e) ... ; and*
> *f) for each source node in said source schema, selecting one of said plurality of node transformation sequences, a selected node transformation sequence, said associated with a least cost based*

> *on an information capacity cost criteria.* (page 88 section entitled "4.1 Value Equivalence")

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Buneman for the benefit of SchemaMatching in view of XEM and Swamy, because to do so would have allowed one to implement a value-based semistructured data model, as taught by Buneman in last paragraph on page 77. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 23, which is dependent upon claim 22, the limitations of claim 22 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *wherein f) in said method further comprises: in a match between a source node and a target node, selecting said selected node transformation sequence to achieve a high quality match, when an associated cost of data loss is less than a second cost of data loss when deleting information contained in said source node, in a first iteration of matching.*

Buneman, though, discloses:

> *wherein f) in said method further comprises: in a match between a source node and a target node, selecting said selected node transformation sequence to achieve a high quality match, when an associated cost of data loss is less than a second cost of data loss when deleting information contained in said source node, in a first iteration of matching.* (page 88 section entitled "4.1 Value Equivalence")

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Buneman for the benefit of SchemaMatching in view of XEM and Swamy, because to do so would have allowed one to implement a value-based semistructured data model, as taught by Buneman in last paragraph on page 77. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 24, which is dependent upon claim 22, the limitations of claim 22 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *matching said source node to said target node having an identical label or synonymous label to achieve said high quality match.*

XEM, though, discloses:

> *matching said source node to said target node having an identical label or synonymous label to achieve said high quality match.* (page 3 section entitled "2.2 The DTD Data Model" discusses node labelling)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of XEM for the benefit of SchemaMatching in view of Swamy and Buneman, because to do so would have allowed a designer to change a DTD without requiring change of underlying XML data, as taught by XEM in top left paragraph of page 2. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 25, which is dependent upon claim 22, the limitations of claim 22 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *wherein f) in said method further comprises: in a match between a source node and a target node, associated cost of data loss is less than a second cost of data loss when deleting source information contained in said source node and adding target information in said target node, in a second iteration of matching.*

Buneman, though, discloses:

> *wherein f) in said method further comprises: in a match between a source node and a target node, associated cost of data loss is less than a second cost of data loss when deleting source information contained in said source node and adding target information in said target node, in a second iteration of matching.* (page 88 section entitled "4.1 Value Equivalence")

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Buneman for the benefit of SchemaMatching in view of XEM and Swamy, because to do so would have allowed one to implement a value-based semistructured data model, as taught by Buneman in last paragraph on page 77. These references were all applicable to the same field of endeavor, i.e., XML programming.

Regarding claim 26, which is dependent upon claim 22, the limitations of claim 22 have been previously addressed.

However, SchemaMatching does not explicitly disclose:

> *wherein f) in said method further comprises: selecting said selected node transformation sequence having the least associated cost of data loss.*

Buneman, though, discloses:

> *wherein f) in said method further comprises: selecting said selected node transformation sequence having the least associated cost of data loss.* (page 88 section entitled "4.1 Value Equivalence")

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Buneman for the benefit of SchemaMatching in view of XEM and Swamy, because to do so would have allowed one to implement a value-based semistructured data model, as taught by Buneman in last paragraph on page 77. These references were all applicable to the same field of endeavor, i.e., XML programming.

Response to Declaration/Affidavit Under 37 CFR 1.131

12. The declarations filed on 10/17/2005 under 37 CFR 1.131 have been considered but are ineffective to overcome the filing date of the Jeong reference ("Induction of Integrated View for XML Data with Heterogeneous DTDs").

The Applicant has submitted declarations signed by the inventors of record (Su, Kuno and Rundensteiner) and supporting documentation, consisting of a paper presented by the inventors at the WIDM 2001 conference in Atlanta.

The Office notes that Jeong reference was originally cited in the non-final communication mailed to Applicant on 11/22/2004. Applicant presented no reason as to why these declarations were not presented at that time. Therefore, the declarations of Su, Kuno and Rundensteiner under 37 CFR 1.131 are insufficient to overcome the Jeong reference because they were not timely presented.

Response to Arguments

13. Applicant's arguments have been fully considered but they are not persuasive.

The arguments presented in Applicant's submission under 37 CFR 1.116 are deemed moot in light of the re-opening of prosecution (and citation of new art).

Conclusion

14. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Non-patent Literature

Su, Hong, et al., "Automating Transformation of XML Documents", WIDM 2001 Powerpoint Presentation, Atlanta, Ga, Nov. 2001, pp. 1-29.

Claypool, Kajal T., et al, "Model Management – A Solution to Support Multiple Data Models, Their Mappings and Maintenance", ACM SIGMOD '01 Industrial Demo, Santa Barbara, CA, May 2001, pp. 1-5.

Lo, Ming-Ling, et al., "XAS: A System for Accessing Componentized, Virtual XML Documents", Proceedings of the 23rd International Conference on Software Engineering, July 2001, pp. 493-502 (plus citation sheet).

Moh, Chuang-Hue, et al., "Re-Engineering Structures from Web Documents", Proceedings of the 5th International Conference on Digital Libraries, June 2000, pp. 67-76 (plus citation sheet) [ACM 1-58113-231-X/00/0006].

Abiteboul, Serge, et al., "Compact Labeling Schemes for Ancestor Queries", Proceedings of the 12th Annual ACM-SIAM Symposium on Software Engineering, Jan. 2001, pp. 493-502 (plus citation sheet).

Fernandez, Mary, et al., "Efficient Evaluation of XML Middle-ware Queries", ACM SIGMOD 2001, May 21-24, 2001, pp. 103-114 [ACM 1-58113-332-4/01/05].

Bonifati, Angela, et al, "Comparative Analysis of Five XML Query Languages", SIGMOD Record, Vol. 29 No. 1, March 2000, pp. 68-79.

Wallace, Malcolm, et al., "Haskell and XML: Generic Combinators or Type-Based Translation?", ICFP '99, Paris, France, Sep. 1999, pp. 148-159 [ACM 1-58113-111-9/99/0009].

US Patent Application Publications

US Patents

15. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Robert M Stevens whose telephone number is (571) 272-4102. The examiner can normally be reached on M-F 6:00 - 2:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Heather R. Herndon can be reached on (571) 272-4136. The current fax phone number for the organization where this application or proceeding is assigned is 703-872-9306. Additionally, the main number for Technology Center 2100 is (571) 272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Robert M. Stevens
Reg. No. 47,972
Art Unit 2176
Date: July 8, 2005

rms

William L. Bashore
WILLIAM BASHORE
PRIMARY EXAMINER
11/10/2005